NASA CR-
160559

Axiomatix

(NASA-CR-160559) EVALUATION OF KU-BAND SPA AND EA-1 SOFTWARE, TASK NO. 2 Interim Report (Axiomatix, Los Angeles, Calif.) 49 p HC A03/MF A01 CSCL 09B

N80-21999

Unclas
G3/61 46790

TASK NO. 2--EVALUATION OF KU-BAND
SPA AND EA-1 SOFTWARE

Interim Report

Contract NAS 9-15795A

Prepared for

NASA Lyndon B. Johnson Space Center
Houston, Texas 77058

Prepared by

James Dodds

Axiomatix
9841 Airport Blvd., Suite 912
Los Angeles, California 90045

Axiomatix Report No. R8003-2
March 20, 1980

1.0 INTRODUCTION

This report covers Task No. 2 of Axiomatix contract number NAS 9-15795A. Task No. 2 is an evaluation of the Ku-band SPA and EA1 software. This report addresses the EA1 software only as no documentation has been made available by HAC on the SPA software. The intent of this task is to augment the HAC documentation and provide a clearer understanding of the software algorithms and programming techniques as well as provide recommendations for improved efficiency and reliability.

The initial phase of this effort entailed a detailed study of HAC documentation, primarily References 1 and 2. Additionally, Reference 3 was needed to correlate software input/output flags and commands with hardware response. Understanding of the software was hampered by the lack of a cross-reference listing of variables, particularly flags used to transfer status between software routines. As part of the task requirement to augment the HAC documentation as well as facilitate our own understanding of the software, a cross-reference table of inter-routine flags has been generated. After the familiarization phase, the software was examined for potential problem areas and possible sub-optimal coding techniques. A bug was discovered in the procedure termination routine and the technique used to store status flags is not as efficient as possible, from both the standpoints of memory usage and processing time. Self-test routines were examined with the intent of providing a more detailed description of MDM and D&C outputs.

1.1 Summary

A minor bug has been found in the procedure termination routine, SWTCH. The alpha and beta position loop integrators are not zeroed, as is claimed. Rather, they are loaded with hex 6060. The fix is to insert an LDI :00 after the LOADR S1, BINTGR+1 at location 0932.

In Section 2.0, we have generated a cross-reference table of status flags. This should aid in tracing the functional activity of software modules, as a description of where a flag is used, set or reset indicates how routines interact. Reconfiguration is one of the more complex executive functions. An example of a reconfiguration is presented in Section 3.0, with a block diagram to indicate the critical

paths. In Section 4.0, we dissect the self-test tasks and include a series of tables which give the MDM outputs after each task as a function of pass or fail. These tables are more detailed than those in [2].

The techniques used to store flags and status information, e.g., storage of up to eight bits per word, requires that each access strip the flag from the word. This means that extra memory and time is required for each access. In Section 5.0, we present an alterate technique for flag storage. Memory usage and processing time are not critical yet, but proposed changes to the software to accommodate feed-forward may require more stringent optimization.

2.0 FLAG CROSS-REFERENCE TABLE

2.1 Introduction

Table 2.1 lists flags, or status bits, which carry information between software routines or between software and external hardware. The flags are listed alphabetically, followed by the module number, as defined in [1] and Table 2.2, and the section within a module where the flag is accessed. A letter following the module and section number indicates the action which is taken with the flag. A flag can be tested (T), set (S), reset (R) or toggled (X). A combination of letters indicates several actions being taken, in the order given. However, if a flag is acted upon several times within one module, only the first of each type is listed. For example, the entry

DETECT 12.6:S,R 20.1:T

indicates that DETECT is set and reset in module 12.6, the set status flag routine, and tested in module 20.1, the self-test task T. Local flags, e.g., flags used within one routine, are not listed. Those flags which are set by hardware input or used to output status information are appended with the name of the appropriate input/output word. These names are indications of the function: OMDM1 is DM output word 1, and PIW1 is parallel input word 1.

Table 2.1. Flag Cross-References

ACQEN	3.1:R,S
ACQINH	12.6:S,R 13.4:T
ANGTRK (IRAD1)	3.1:T 18.5:T 20.1:T
ASTOP	2.5:T,S,R 14.1:T
ATIME	2.1:T,S
ATTEN1 (POW5)	2.3:R 18.5:R 19.3:R 20.1:R 21.1:S
ATTEN2 (POW5)	2.3:R 18.5:R 19.3:R 20.1:R 21.1:R
AUTO (PIW1)	3.1:T, 5.1:T 12.3:T 12.6:T 13.1:T 13.4:T
BOOM (PIW4)	3.1:T
BSTOP	2.5:T,R,S 14.1:T
BTIME	2.5:T,S
COASTA (PIW2)	3.1:T 12.6:T
COMON (PIW4)	3.1:T 12.6:T
COMSTB (PIW4)	3.1:T 12.6:T
CTIME	2.3:R 16.2:R,T,S
DATAGO	12.6:S,R 13.1:T
DATAP (PIW2)	12.6:T 13.4:T
DATEST	2.3:R 13.1:T 18.1:R 21.1:S,T
DETECT	12.5:S,R 13.1:T 20.1:T
DOPOS	1.4:T 18.1:R 19.1:S,R 19.2:R,S 22.1:S,R
DOWN (PIW3)	15.1:T
DPLYF	1.2:T 14.1:R 15.3:S
DWELL	5.1:S,R 6.2:T
EAST (IMDM1)	3.1:T
EBW2	0.1:T 2.1:S 3.1:S
EBW3	0.1:T 2.1:S 3.1:S
EDW1	3.1:S,T
EDW2	2.3:T 3.1:S
EDW3	2.3:R 3.1:R,S,T
EVEODD	0.1:R,T 0.2:R 2.5:X 15.2:T
EVW1	0.1:T 0.2:S 2.1:S 3.1:S
E1TEST	2.3:R 13.1:T 18.1:R 21.1:S,T
E2TEST	2.3:R 13.1:T 18.1:R 21.1:S,T
FAST (PIW3)	15.1:T
FSTSLO	3.1:T 12.0:T 13.4:T 15.1:S,R

Table 2.1. Flag Cross-References (Cont'd)

GAIN3	12.1:R,S
GPCACQ (PIW1)	3.1:T 7.2:T 12.3:T 12.6:T 13.1:T 13.4:T
GPCDES (PIW1)	1.2:S 3.1:T 7.2:T 12.3:T 12.6:T 13.1:T 13.4:T
GPCLS	12.3:T
INCDEC	5.1:R,T,S 6.1:T
INIT	0.2:S 2.1:S,R 3.1:T
INSTAB	1.2:R 2.3:R 7.1:T 7.4:T 12.3:R 19.1:R 19.2:R
KOUNTL	5.1:S,T
LAMP	18.1:R 21.1:S 21.3:T
LATCH (PIW2)	3.1:T,R 12.5:R
LEFT (PIW3)	15.1:T
LINEAR (POW1)	12.5:R,S
MAIN	13.4:R,T
MAMINI	5.1:T,S,R 13.4:T
MANUAL (PIW1)	2.6:T
MCOMP	2.3:R
MFIRST	1.4:T 2.1:T,S 3.1:R,S 5.1:T 14.1:T 15.1:T 16.2:T
MINIST	0.8:T 2.3:R 3.1:S 5.1:T,R
MINI2 (IRAD2)	5.1:T
MODE	0.2:S 2.1:T 3.1:T,S
MOTON (POW4)	2.2:R 2.3:S 14.1:S,R 19.3:R 20.1:R
NEWA (PIW4)	13.1:T 14.1:T
NEWB (PIW4)	13.1:T 14.1:T
NEWMDM (PIW1)	0.1:T 0.3:T 0.4:T 3.1:T 7.1:T
OPER (PIW2)	13.1:T 19.3:T
OUTFLG	1.4:T,R 2.3:R 18.1:S 18.4:S 18.5:S 19.1:S 19.2:S 19.3:S 20.1:S 21.1:S
PFLAG	7.1:R,S 7.2:T 7.3:T,S
PNTTRK	2.1:S 3.1:S,R
PSON (POW1)	2.1:S 3.1:R
PTIME	2.3:R 3.1:T,S
RADACT (PIW1)	12.5:T 13.4:T
RADCOM	1.1:T 3.1:T 5.1:T 12.5:T 12.6:S,R,T 13.1:T 13.3:T 13.4:T 16.2:T
RADON (PIW1)	3.1:T 12.6:T

Table 2.1. Flag Cross-References (Cont'd)

RADSTB (PIW4)	3.1:T 12.6:T
RDETCT (IRAD2)	12.6:T
RGOOD (IRAD1)	13.1:T 18.5:T,S 19.3:R 20.1:T 22.1:R
RIGHT (PIW3)	15.1:T
ROPER (IRAD1)	18.4:T
RRGOOD (IRAD1)	13.1:T 18.5:T,S 19.3:R 20.1:T 22.1:R
RTRACK (IRAD2)	12.6:T
SCANNG	2.3:R 3.1:T 5.1:T,S,R 13.1:T 13.4:T 21.3:S
SCANRC	5.1:T,R,S 6.2:T
SCWARN	2.4:R 4.2:R,S 13.1:T 21.3:S
SELF	2.3:R 13.1:T 18.1:S
SIGEN (POW5)	2.3:R 18.2:S 18.5:S 19.3:R 20.1:S 21.1:S
SIGNAL	12.6:R,S,T 13.4:T
SIGPC (IMDM1)	3.1:T
SIMAN (PIW2)	3.1:T
SKEY (PIW3)	11.6:T 15.3:T
SLEINT	15.1:S 15.2:T,R
SLEWNG	2.3:R 3.1:T 7.1:T 12.3:T 12.6:T 13.4:T 15.1:S,T,R
SLOW (PIW3)	15.1:T
STCON1 (POW5)	2.3:R 18.5:R 19.3:S 20.1:R 21.1:R,S
STCON2 (POW5)	2.3:R 18.5:S,R 19.3:S 20.1:S 21.1:S,R
STCON3 (POW5)	2.3:R 18.5:S 19.3:S 20.1:S 21.1:S
STCON4 (POW5)	2.3:R 18.5:R 19.3:R 20.1:R 21.1:R
STEST (IMDM1)	3.1:T
STON (POW5)	2.3:R 18.5:S 19.3:S 20.1:S 21.1:S
STOWM (POW3)	2.1:R 14.1:R,S
STUNST	3.1:S,R 14.1:T
STWAIT	2.3:R 18.1:S 18.S:T 18.5:S 19.1:S 19.2:S 19.3:S 20.1:S 21.1:S 22.1:R
SYSTST	2.3:R 13.1:T 18.1:R 21.1:S
TARGET	2.3:R 18.4:T 18.1:S 19.3:R 20.1:S,R
TDEAST	3.1:T,S,R 13.1:T 13.4:T
TDWEST	3.1:R,T,S 13.1:T 13.4:T
TIMOUT	2.3:R 3.1:T,S,R 5.1:T

Table 2.1. Flag Cross-References (Cont'd)

TRACK	12.6:S,R 13.1:T 20.1:T 21.1:T 21.3:S
TRKING	2.3:R 3.1:T 12,6:T 16.2:S
UNLOCK (POW3)	14.1:S 15.3:R
UNSTOD	3.1:T 14.1:T
UP (PIW3)	15.1:T
WAITC	0.1:T,R 0.2:S 2.5:S
WEST (IMDM1)	3.1:T
XMIT (PIW4)	3.1:T
ZERDIS	2.1:S 2.2:S 2.3:R 13,1:T
ZONEI	3.1:T 5.1:R 7.2:S,R 12,3:T 15,3:T 19.1:R,T 19.2;R,T 22.1:R
ZONEO	3.1:T 7.2:S,R 12.6:T 15.3:R,S

Table 2.2. Software Module Index

0.1	INTERRUPT/EXEC RTN
0.2	POWER UP RTN
0.3	INITIALIZE PROC
0.4	POINT PROC
0.5	IDLE PROC
0.6	PROC TERM PROC
0.7	SLEW PROC
0.8	SCAN PROC
1.1	TRACK PROC
1.2	DEPLOY PROC
1.3	RECOVER PROC
1.4	SELF TEST PROC
2.1	INITIALIZE ROUTINE
2.2	IDLE ROUTINE
2.3	PROC TERM ROUTINE
2.4	OUTPUT STATUS ROUTINE
2.5	WAIT ROUTINE
3.1	CONFIGURE ROUTINE
4.1	ANGLE RATE XFORM
4.2	OBSCURATION CALC
5.1	SCAN1 ROUTINE
6.1	SCAN2 ROUTINE
6.2	SCAN3 ROUTINE
6.3	SCAN4 ROUTINE
7.1	POSITION LOOP, MOD1
7.2	POSITION LOOP, MOD2
7.3	POSITION LOOP, MOD3
7.4	POSITION LOOP, MOD4
11.6	SHORT SHUTTLE TO GIMBAL TRANSFORM
12.1	ENCODER ROUTINE
12.2	ANALOG ROUTINE
12.3	INERT ROUTINE
12.4	INPUT DISCRETE ROUTINE
12.5	OUTPUT DISCRETE ROUTINE
12.6	SET STATUS FLAGS ROUTINE
13.1	MDM/D&C OUTPUT ROUTINE
13.3	INPUT LRU SERIAL ROUTINE
13.4	OUTPUT LRU SERIAL ROUTINE
14.1	DEPLOY ROUTINE
15.1	SLEW GENERATION ROUTINE
15.2	SLEW INTERPOLATION ROUTINE
15.3	DEPLOY ROUTINE

Table 2.2. Software Module Index (Cont'd)

16.2	[illegible]UTINE
18.1	SELF TEST INITIALIZATION ROUTINE
18.2	SELF TEST SEQUENCER
18.3	SELF TEST TASK2 CPU
18.4	SELF TEST TASK3 POWER FORM
18.5	SELF TEST TASK8 ANGLE TRACK
19.1	SELF TEST TASK4 INITIALIZE ANTENNA
19.2	SELF TEST TASK5 ANTENNA SERVO
19.3	SELF TEST TASK6 TRANSMITTER POWER CHECK
20.1	SELF TEST TASK7 RANGE AND RANGE RATE CHECK
21.1	SELF TEST TASK9 RECEIVER SENSITIVITY CHECK
21.2	SELF TEST TASK10 COMPILE TEST RESULTS
21.3	SELF TEST TASK11 LAMP TEST
22.1	SELF TEST PAUSE ROUTINE

3.0 EXAMPLE OF RECONFIGURATION

3.1 Introduction

In order to facilitate an understanding of the software and to bridge the gap between the HAC-supplied listing and block diagrams, a hybrid block diagram, Figure 3.1, was generated to provide an example of a reconfiguration. Reconfiguration is probably the most complex of the executive functions, and an understanding of this function will aid in understanding the executive structure.

For purposes of the example, a reconfiguration from IDLE to SELF TEST is illustrated. Four interrupt cycles are covered which demonstrate the function of the termination procedure and the initialization routine for self-test. Each interrupt cycle is a separate diagram. Each block is labeled with the absolute hexadecimal address of the routine in ROM and contains descriptive text as to the function to be performed. Action on variables or flags is underlined in the blocks. Blocks are grouped within modules in order to ease cross-referencing with the software listing.

3.2 Description of Reconfiguration

Initially, in the IDLE routine, executive data word 1 equals MIDLE (=2), executive branch word 1 (EBW1) = 12, EBW2 = 15 and EBW3 = 18. The latter are the absolute memory locations of the long branches to the 0.1-second sequence, the even and odd sequences of the idle procedure. The 0.1-second sequence tests the even/odd flag, EVEODD, branches to EVEN, executes INDIS (input discrete), executes MSDIN (input MDM), then executes CONF, the configure routine on module 3. The configure routine tests the status flags, notes that STEST is set, and sets EDW1 = MSELF. Configure then notes that MODE is not equal to EDW1; hence, a new procedure is to be initiated. Since the current mode is not the termination procedure, MODE and EDW4 are set to 8 to indicate that the termination procedure is to be executed, MFIRST is set to 3 for the first pass through and EBW1 is set to 51, EBW2 to 54 and EBW3 to 57.

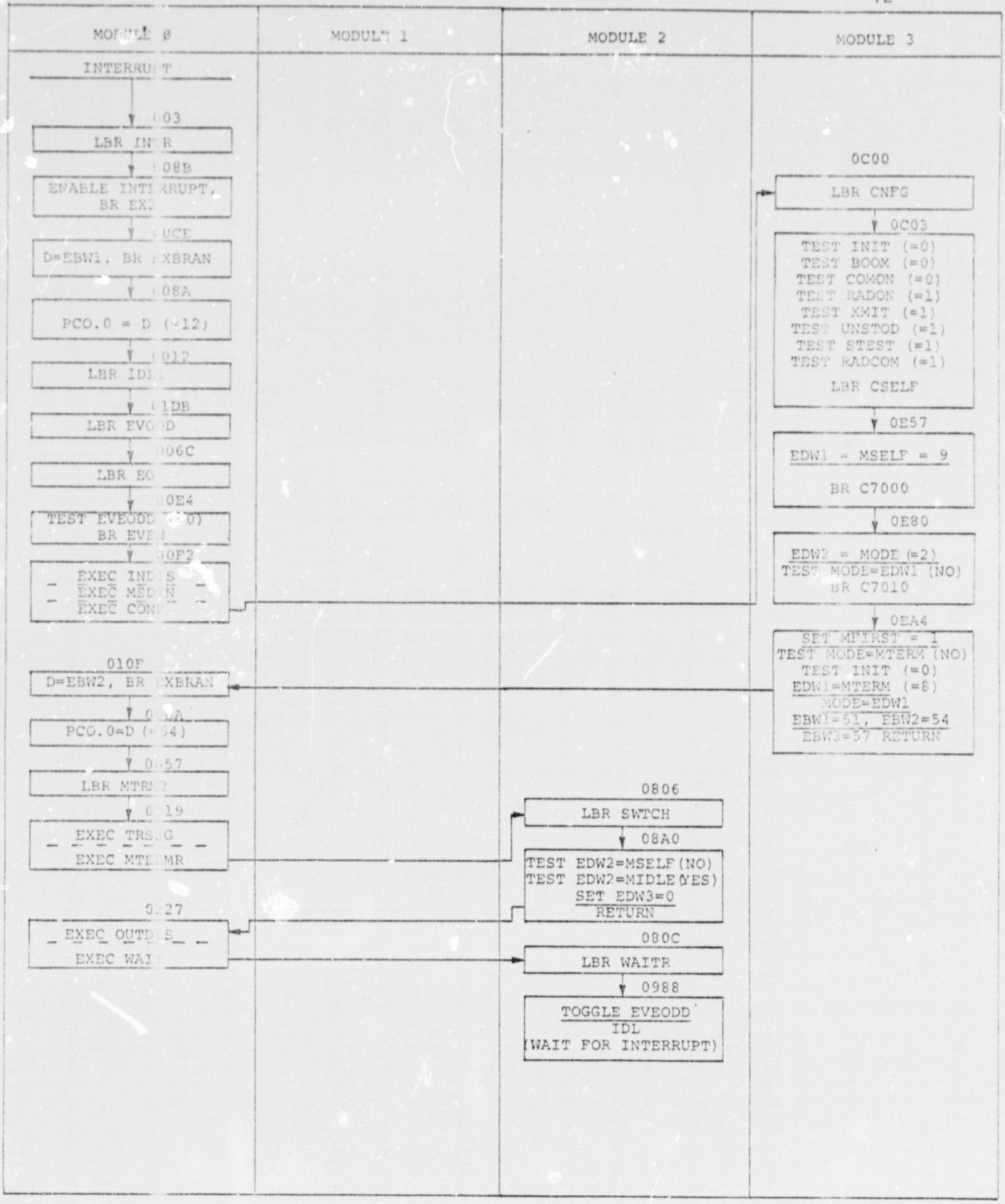
After configure, the executive takes the even leg of the procedure, which is now the termination procedure, MTRM2. This procedure executes TRS2G, the Shuttle-to-gimbal transform, then executes the

termination routine, MTERMR, which sets EDW3 equal to 0, and returns to the procedure. The termination procedure outputs discrete data, then executes WAIT, which toggles the even/odd flag to odd and idles for the next interrupt.

During the next interrupt, the executive branches to the termination procedure odd leg, which executes TRG2S, transform gimbal-to-Shuttle, executes MSDOUT, MDM output, and finally executes WAIT, which toggles the even/odd flag back to even.

During the third interrupt, the self-test procedure is executed for the first time. Configure tests EDW1 and MODE for equality. Since they are not equal and the current MODE is 8, the termination procedure, configure again sets MFIRST, the first pass flag, to 1 and configures for self-test by setting the executive branch words. Upon return to the executive, a long branch to the even leg of the self-test procedure is executed. The self-test initialization is rather busy. With MFIRST equal to 1, STST2 executes self-test initialization, STINIT, which does a long branch to BEGIN. Various flags and buffers are initialized, including the step counter, STPCNT, and LAMP. BEGIN returns to the self-test procedure which inputs LRU serial data, then executes the self-test task sequences. With STWAIT equal to 1 and STPCNT equal to 0, a ten-second pause is initiated and WAIT is executed.

Finally, the fourth interrupt results in the execution of the odd leg of the self-test procedure. After the gimbal-to-Shuttle transform, task 11 is executed. Since LAMP was initialized to 0, the D&C lamps are not lit. Task 11 returns to the odd leg of the self-test procedure, which outputs discrete, outputs LRU serial, and outputs MDM data since OUTFLG was set to 1 by STINIT. WAIT is executed and the even/odd cycle repeats. A more detailed discussion of the self-test routines is covered in the next section.



INITIAL CONDITIONS: EDW1 = MIDLE = MODE = EDW2 = 2,
EBW1 = 12, EBW2 = 15, EBW3 = 18
EVEODD = Ø (EVEN)

FIGURE 3.1. 0.1-SECOND INTERRUPT; CONFIGURE FROM IDLE TO SELF-TEST

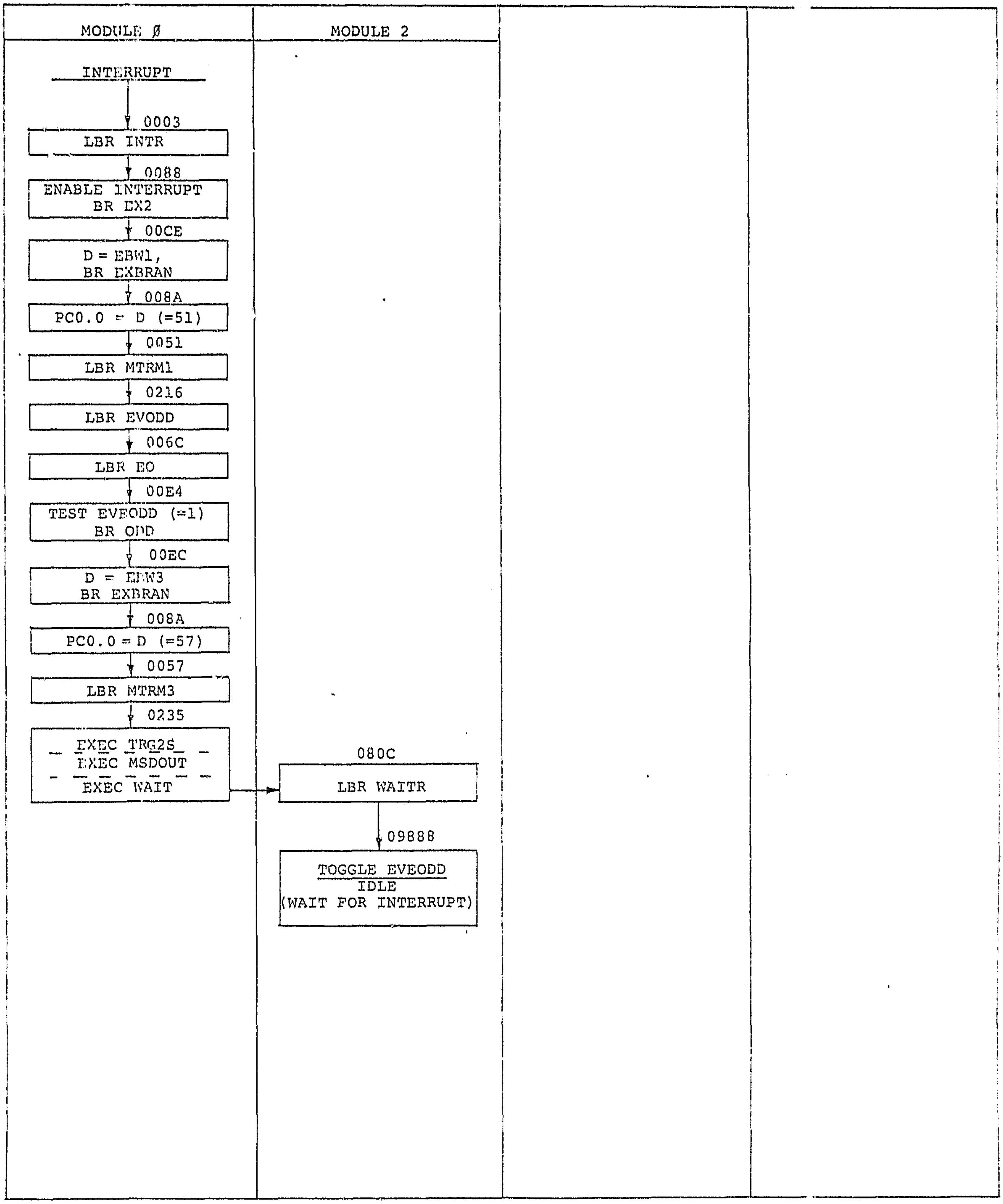


Figure 3.1 (Cont'd)

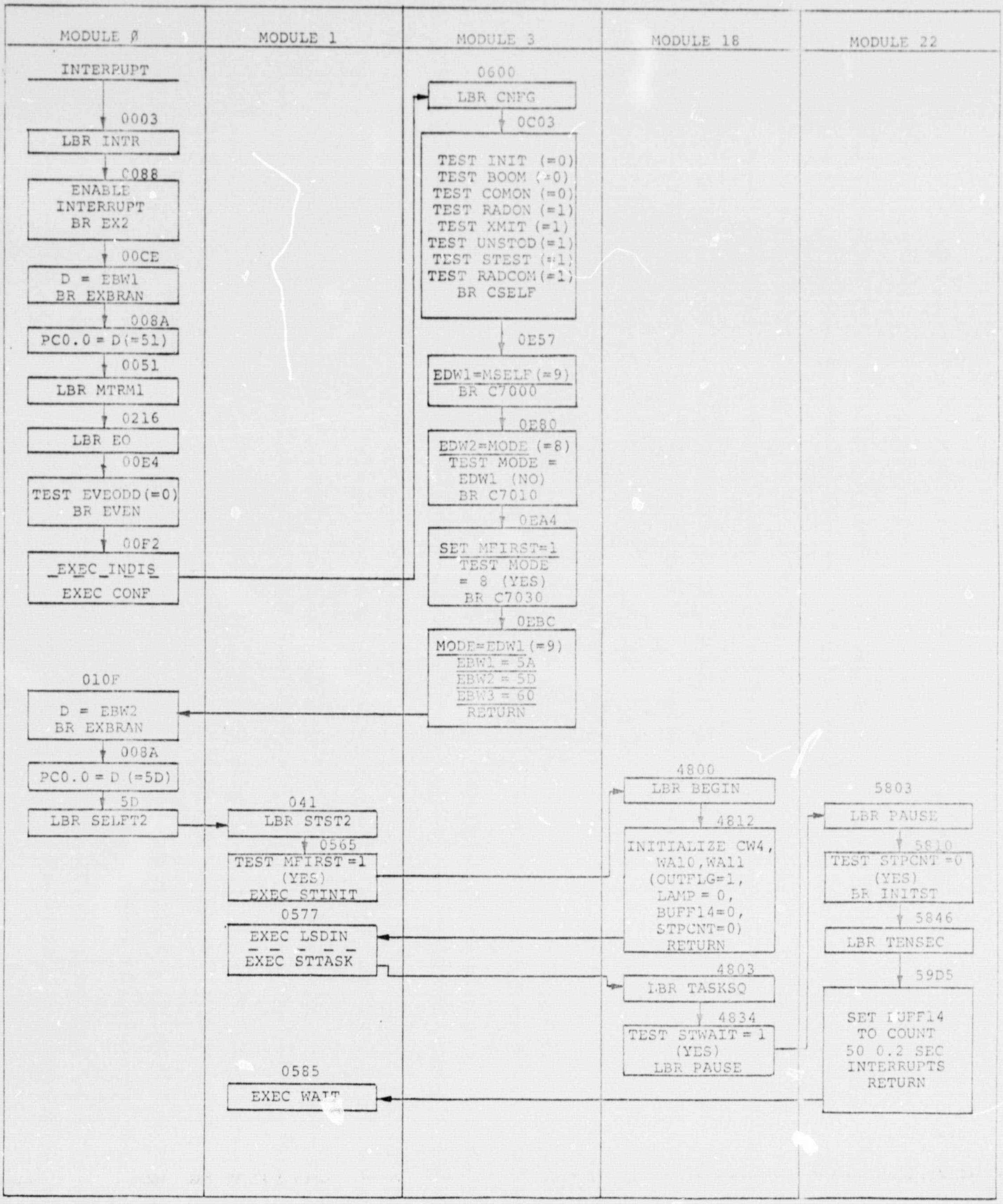
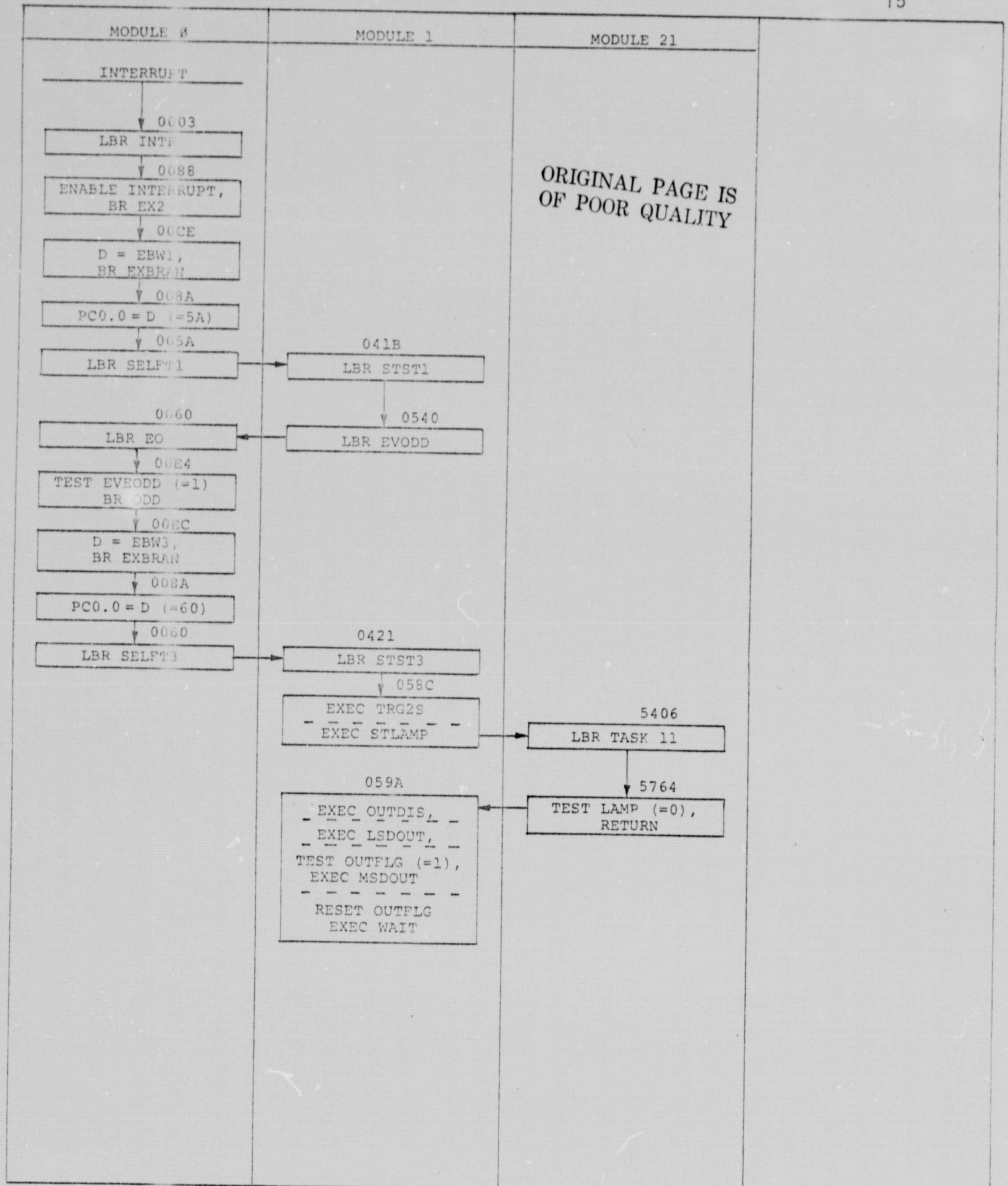


Figure 3.1 (Cont'd)



ORIGINAL PAGE IS OF POOR QUALITY

Figure 3.1 (Cont'd)

4.0 SELF-TEST TASKS

4.1 Introduction

Operation of the self-test portion of the EA-1 software is examined with emphasis on the outputs generated at the completion of each self-test task. The intent is to provide a reference to augment the H&C description of the outputs, as well as a guide to modification of the software, should it be considered desirable to change the output format. As an example, the present implementation does not provide an unambiguous indication as to which task is currently in progress during self-test. This would be changed by modification of a few memory locations, as described in subsequent paragraphs.

Self-test tasks are discussed in chronological order, with key routines and parameters referenced to their absolute memory locations. MDM outputs are summarized at the conclusion of this section in Tables 4.2 through 4.9.

The self-test routines make extensive use of macros specific to self-test. These macros are explicitly defined in the self-test portion of [1]. Macro CONFIG A,B loads parallel input word one (PIW1), bits 4 through 8, with the value A and bits 1 through 5 and 7 with the value B. Macro RANG A,B,C loads values A,B,C into EA2 serial input words IRAD3, IRAD4 and IRAD5. Similarly, RANGRT A,B loads A and B into IRAD6 and IRAD7. These two macros allow the software to output specified values of the range and range rate to the MDM's and D&C using the MDM output routine, MDMO (340F), which transfers data from the EA2 input to the MDM and D&C output registers. ANGRTE A,B,C,D transfers the specified data to the MDM output registers OMDM17 through OMDM20 in order to force display of angle rates. Macro POSIT A,B,C,D stores the data A-D in IMDM3 through IMDM6, the MDM input registers. This allows the software to position the antenna to roll specified by A,B and pitch specified by C and D, overriding angle designates read in from the MDM's. Macro DAVOLT A,B outputs A and B to the digital-to-analog converter. This voltage is used by the self-test multiplexer/comparator to determine if specified signals exceed the voltage threshold. Macro ANDATA A,B,C,D is similar in that it loads A and B into the analog output registers ADAM and ADAL for setting the alpha axis rates and C and D into BDAM and BDAL to set the beta axis rates. Actual output is done using the subroutine ANALOG, also called SERVO, at location 3084.

The self-test MUX selects one of four analog inputs, as defined in Table 4.1, and compares the voltage with the output of the alpha axis rate D/A converter. During this test, the motors are turned off. The difference of the two analog voltages is sent to the processors event flag EF3, for test.

In general terms, each self-test task uses a sequence counter, SEQUEN, zeroed by the previous task, to govern which phase of the task to execute. Typically, with SEQUEN equal to 0, the task initializes itself and sets a flag, STWAIT, to pause 10 seconds for display of MDM and I&C output from the last sequence of the prior task.

All self-test tasks, with the exception of the lamp test, are executed during the even cycle of the self-test procedure. The first pass through the self-test procedure (MFIRST = 1) will result in the execution of the initialization routine. This is entered via

```
0570                EXEC       PC1,STINIT,
4800                LBR        BEGIN
4812     BEGIN      LOADR      IOP,CW4 .
```

BEGIN, the initialization routine for the self-test procedure, resets RAM WA10, WA11 and sets various flags:

```
SELF = 1            indicate in self-test
TARGET = 1          turn on test target
DATEST = 0          reset status flags
E2TEST = 0
E1TEST = 0
SYSTST = 0
STWAIT = 1          set for 10-sec pause
OUTFLG = 1          set to output MDM.
```

The self-test task sequencer, TASKSQ, uses the variable STPCNT to control the sequence of self-test routines. STPCNT is reset to 0 by virtue of being located in WA10. BEGIN returns to the self-test procedure via SET PCØ.

The self-test tasks are entered via

```
057E                EXEC       PC1,STTASK
4803                LBR        TASKSQ
4834     TASKSQ     SETT       IOP,SIGEN.
```

Initially, with STWAIT set, TASKSQ branches to PAUSE, which implements a 10-sec wait. At the conclusion of 10 seconds, PAUSE resets STWAIT to enable the first self-test routine. During the final pass through PAUSE,

Table 4.1. Self-Test MUX Control

Function	STCON1	STCON2	STCON3
BETA ERROR	0	0	1
SIGNAL STRENGTH	0	1	1
ALPHA ERROR	1	0	1
XMIT RF POWER	1	1	1

STPCNT is tested and the routine branches to INITST (location 5846), which initializes the MDM output. Data-good flags are reset and MDM words are set as follows:

RANG	(range) set to all 1's
RANGRT	(range rate) set to all 1's
ANGRTE	(angle rate) set to 0's.

Values of the angles are not modified by INITST.

During the first pass through the odd leg of the self-test procedure, MDM output is initiated since OUTFLG had been initialized to 1.

4.2 Tasks 1 and 2

At the conclusion of the first 10-sec pause, TASKSQ tests STPCNT (=0) and branches to TSKCPU,

```
484E            LBZ      TSKCPU
5800            LBR      TASK1
5806    TASK1   LOADR    TO,STPCNT
```

Task 1, the CPU self-test, has been eliminated. The net effect of TASK1 is to set STPCNT equal to 1 and return. Thus, during the next pass through the even leg of the self-test procedure, TASKSQ will branch to TSKPCS, the PROM self-test,

```
4854            LBA      TSKPCS
4809            LBR      TASK2
489C    TASK2   LOADRS   IOP,SEQUEN.
```

At the conclusion of the PROM test, the PROM check sum flag is set and STPCNT is set to 2 (task 3) if the test is successful. Otherwise, the flag is reset, STPCNT is set to go to task 10, and self-test will be prematurely terminated. New MDM output is not initiated after PROM test.

4.3 Task 3

During the next pass through the even leg of the self-test procedure, the EA2 power form task is executed:

```
484A            LBA      TSKPFC
480C            LBR      TASK3
4914    TASK3   LOADRS   IOP,IRAD1
```

Task 3 first resets the four data-good flags, sets range and range rate to all 1's and sets the angle rates to 0. MDM output is not conditioned

on the results of the power form test. The radar operate bit from the EA2 is tested and, if true, PFC flag is set to 80. STPCNT is set to 3 for task 4, OUTFLG is set to initiate MDM output, and control is returned to the even leg of the self-test procedure. MDM (and D&C) data will be output during the next pass through the odd leg of the self-test procedure. However, STWAIT has not yet been set, and the first pass through Task 4 will be executed without undergoing a 10-sec pause.

4.4 Task 4

The first pass through Task 4, which initializes the antenna to zenith, again resets the data-good flags, sets range and range rates to all 1's and the angle rates to 0's. Mode is set to GPC designate. SEQUEN is tested to determine if this is the first or a subsequent pass. SEQUEN is 0 during the first pass, having been cleared as part of work area 10 (WA10+17). STWAIT is set to enable the 10-sec pause and control is returned to the self-test procedure. After the first pass through Task 4, the task sequencer will branch to PAUSE, which tests STPCNT to determine the current task. STPCNT is 3, which causes PAUSE to branch to ZEN1 (5883). Using the POSIT macro, ZEN1 loads 0's into the angle designates IMDM3 through IMDM4 and branches to POSLOP (5983). POSLOP sets the position loop flag, DOPOS, and starts the 10-sec pause. With DOPOS set, the position loop routine is executed every 100 ms prior to executing either the even or odd leg of the self-test procedure. At the conclusion of the 10-sec pause, DOPOS is rest, as is the ZONEI inner zone flag.

During the second and subsequent calls to Task 4, SEQUEN is set and TASK4 branches to DES (4C7B). DES again sets the angle designates to 0, tests ZONEI to determine if the antenna is within 0.3° of the designate and sets the position loop flag. BUFF5, initially cleared as part of work area 10, is tested for excessive time to reach zenith. Ten seconds after the end of the 10-sec pause are allowed for the antenna to reach zenith. If the limits are exceeded, DES branches to ABORT (4C00), which sets STPCNT to skip Tasks 5 and 6 and sets OUTFLG to initiate MDM output. Measured pitch and roll angles are stored in the MDM and D&C output words, OMDM5 through OMDM8, and ODC1 through ODC6, by the gimbal-to-Shuttle transform routine G2S1 (2C0C), executed prior to MDM

output. If the antenna is successfully positioned at zenith, DES branches to READY (4CAA), which resets the position loop flag. The ZONE1 inner zone flag, the sequence counter SEQUEN, sets STPCNT for Task 5 and sets OUTFLG for MDM output. In either case, no flag is set to indicate the results of Task 4. However, if Task 4 fails and Tasks 5 and 6 are not executed, ASD and TPLF will be 0 since they are in WA11 and an EA1 or DA error will be indicated. ASD (WA11+4) is set by Task 5 and TPLF (WA11+5) is set by Task 6.

4.5 Task 5

Task 5 (4CDE) tests the ability of the servos to point the antenna to designated positions within a specified time. This task is divided into four sequences--0 through 3. Sequence 0 is initialization, sequence 1 points to pitch = 30°, roll = 0°; sequence 2 points to pitch = 30°, roll = -30°; and sequence 3 points back to zenith. There are three possible results of the Task 5 tests: unconditional pass if the antenna reaches designate within 7 sec, conditional failure if it reaches designate within 7 to 10 sec in pitch or roll, and unconditional failure if it fails to reach designate within 10 sec. During sequence 0, STWAIT is set for the 10-sec pause and the sequence number is set to 1. The 10-sec pause routine tests STPCNT (4) and branches to ANT (5899), which tests the sequence number, SEQUEN. With SEQUEN equal to 1, ANT branches to ZEN1 (5883), which positions the antenna to zenith prior to the pitch test. If the test passes or fails conditionally at the conclusion of sequence 1, pitch output should read 30° and roll 0°. If the test fails unconditionally, Task 5 is terminated, STPCNT is set to go to Task 7, OUTFLG is set to initiate MDM output, but STWAIT is not set for the 10-sec pause--this is set in sequence 0 of Task 7. The 10-sec pause prior to the roll test (Task 2) designates the antenna to the pitch position attained in Task 1 and the 10-sec pause prior to the zenith test (Task 3) designates the antenna to the roll position from Task 2. As in sequence 1, the designate positions are read out during the 10-sec pause following sequences 2 and 3.

4.6 Task 6

Task 6 (4EAC), the transmitter power level test, consists of three sequences. The first is initialization, the second tests the DA operate bit, and the third verifies that the transmitted power exceeds threshold. The initialization sequence sets the range and range rate to all 1's and the angle rate to all 0's, as well as resetting the data-good flags, as in prior tasks. Estimated range is set to 10,250 ft in IMDM7 through IMDM10 (6167-616A) for output to the EA2. STWAIT is set for the 10-sec pause, which again sets an estimated range of 10,250 ft. After the first 10-sec pause, Task 6 branches to CHKOPB (4F41), which tests the DA operate bit. The logic of the DA operate bit is depicted in Figure 4.1. If the operate bit is true, bit 8 of TPLP (61E5) is set. The transmitter is enabled by the commands:

4F4F	RESETS	IOP,SIGEN
4F58	RESETS	IOP,ATTEN1
4F61	RESETS	IOP,ATTEN2.

Transmitter power level is selected by the following commands:

4F6A	SETS	IOP,STCON1
4F71	SETS	IOP,STCON2
4F78	SETS	IOP,STCON3.

Comparator voltage threshold is set by the macro:

4F8F	DAVOLT	TRP1,TRP2 .

OUTFLG is set, STWAIT is set and control is returned to the even leg of the self-test procedure. The MDM output, pass or fail, will be the same as sequence 3 of Task 5--range and range rate all 1's, angle rate all 0's, pitch and roll 0's. After the second 10-sec pause, TASK6 branches to CHKVLT (4FBA), which tests the status of external flag 3, EF3, which will be true if the transmit power exceeds threshold. OUTFLG is set and MDM output will remain the same as sequence 1, whether the test passes or fails. STPCNT is set to 6 for task 7 and OUTLFG is set for MDM output.

4.7 Task 7

Task 7 consists of four sequences: initialization, radar active, test target off, and radar passive. During the initialization sequence, mode is set to manual by the macro

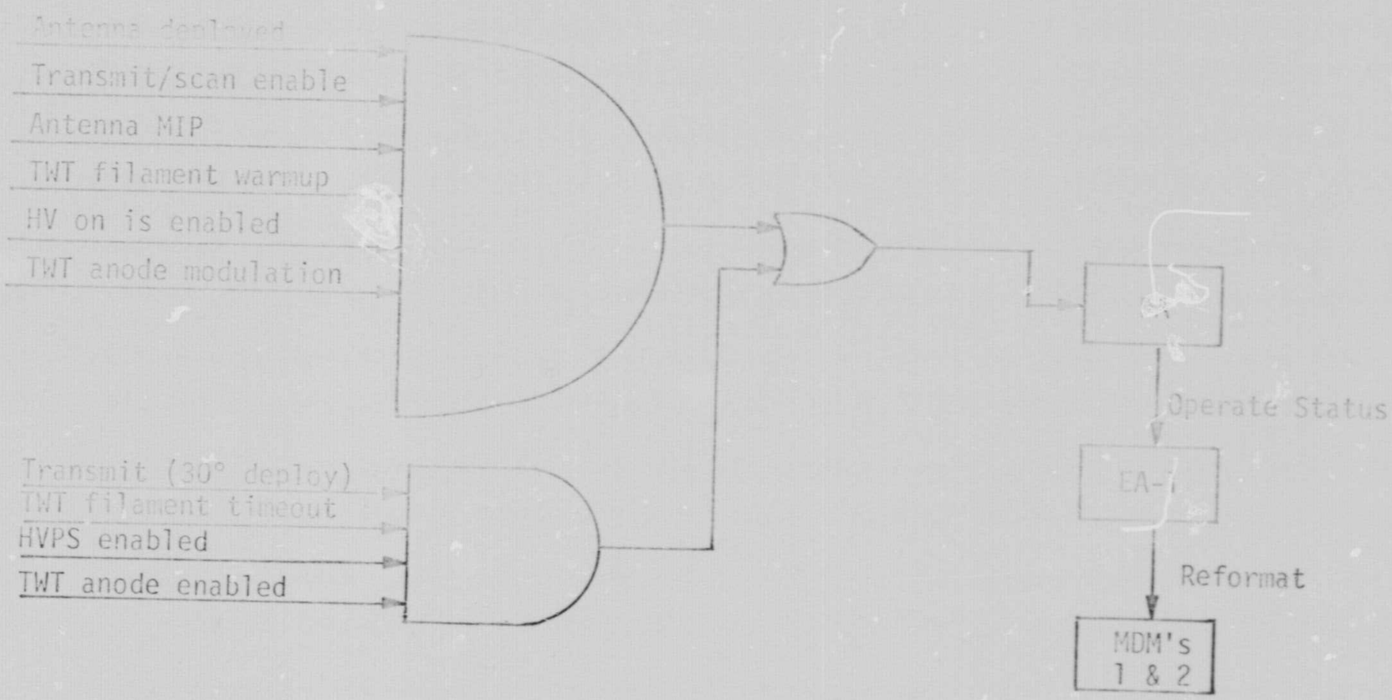


Figure 4.1. Operate Signal Logic Diagram

500C TASK7 CONFIG :88,:40.

Angle and angle rate flags are cleared, estimated range of 10,250 ft is stored in the MDM input buffer, and TARGET is set. The TARGET flag is sent to the EA2 during LRU serial output to turn on the test target. POW5 is set to 8D, which sets the test dipole to -17.5 dBm. SEQUEN is set to 1, gimbal motors are turned off, and STWAIT is set to provide the 10-sec wait for display of Task 6 results. The pause routine branches to RRWAIT (58DC), which again sets the estimated range to 10,250 ft and configures the system to manual-active mode. After the first 10-sec pause, TASK7 branches to ACTIVE (507A). Bit 8 of ACTFLG (61E6) is set if the track status flag is true and bit 7 is set if the angle track flag is false. ACTIVE then executes a call to RNGACT (5295) to test if the measured range is within limits. The range limits are 5570 and 5070 ft. A 1 is returned in BUFF7 if range is within limits. ACTIVE in turn sets bit 6 of ACTFLG if the range is valid and sets bit 5 if the range-good flag, RGOOD, is true. ACTIVE then calls RRATE (52C2) to determine whether or not the range rate data is within limits of ±3 ft/s. If the high-order bits of range rate (IRAD6) are 0, 3 ft/s (3C) is subtracted from the low-order bits. A negative or 0 result indicates that range rate is less than 3 ft/s. Conversely, if the high-order bits are all 1's, -3 ft/s is subtracted from the low-order bits. A negative result indicates that the velocity is greater in magnitude than the limit. Passing the limit test results in bit 4 of ACTFLG being set, and bit 3 is set if RRGOOD, the range rate good status flag, is true. Signal strength from the test target is compared with thresholds S1 and S2. If greater than S1, bit 2 of ACTFLG is set and, if less than S2, bit 1 is set. The test target is turned off and OUTFLG and STWAIT are set. Note that output to the MDM and D&C will reflect the actual range and range rate computed by the EA2, as well as the state of the data-good flags. That is, the range and range rate flags will be 1 unless the test fails. Pitch and roll will be 0° if Tasks 5 and 6 are successful. Otherwise, pitch and roll will reflect the actual pitch and roll as computed by the gimbal-to-Shuttle routine, reflecting the measured gimbal angles. Angle rates will be 0 since they had been set to 0 by previous routines and the angle rate routine has not been called.

The third phase of Task 7, sequence 2, tests the detect and track flags with the target off. If DETECT is false, bit 8 of OFFRRR (61E7) is set and, if TRACK is false, bit 7 is set. The test target is turned back on, SEQUEN is set to 3 and OUTFLG and STWAIT are set. Range and range rate data-good flags to the MDM will be 0 if the test passes, as will angle and angle rate-good flags. Range and range rate will be 0 if the test passes since, from [4], EA2 serial output is forced to 0 unless tracking. Pitch and roll will be as in the prior sequence and angle rate will be 0.

Sequence 3 of Task 7, radar passive, starts at PASIVE (5194). Bit 8 of PASFLG (61E8) is set if the track flag is true and bit 7 is set if ANGTRK is false. PASIVE then calls RNGPAS (526C) to check if the measured range is 10,240 ± 250 ft. Bit 6 of PASFLG will be set if this test is successful. Bit 5 is set if the range data-good flag is true. PASIVE then calls RRATE to determine if the measured range rate is 0 ± 3 ft/s. If successful, bit 4 of PASFLG is set and bit 3 is set if the range rate data-good flag, RRGOOD, is true. Signal strength is compared with thresholds S3 and S4. If it is greater than S3, bit 2 of PASFLG is set and, if less than S4, bit 1 is set. STPCNT is set to 7 for Task 8 and OUTFLG is set for MDM output. Range and range rate data-good will be 1 if the test passes, range will be measured range, 10,240 ± 250 ft, and range rate will be between ±3 ft/s if the test passes. Angle rates will be 0 and pitch and roll will be as in the prior sequences. Note that tasks subsequent to Task 7 do not explicitly turn on the test target but rely on TARGET being set by TASK7. This would be important if the tasks were to be shuffled for some reason.

4.8 Task 8

Task 8 (496D) has four parts: initialization, data-good flag test, alpha error signal within limits, and beta error signal within limits. During initialization, the system is configured to auto-track by the macro:

```
496D     TASK8     CONFIG     :40,:0.
```

Estimated range is set to 10,250 ft and angle and angle rate good flags are set to 1. SEQUEN is set to 1, POW5 is set to 8D to provide -17.5 dBm at the test dipole, and STWAIT is set for the 10-sec pause to display the

results of Task 7, sequence 3. The task 8 pause merely reestablishes the estimated range of 10,250 ft. Sequence 1 of Task 8, starting at RFLG (49C5), sets bit 8 of ANTR (61E9) if range rate data-good is true and sets bit 7 if range data is good. POW5 is set to 95 to command the self-test MUX to measure the alpha error signal with the test dipole at maximum power. OUTFLG and STWAIT are set, and control is returned to the even leg of the self-test procedure. MDM output will consist of range and range rate data-good flags equal to 1 if the test passes. Angle and angle rate data-good flags are set by software and will be 1 unconditionally. Range and range rate will be that provided by the test target, presumably close to 10,240 ft and 0 ft/s. Angle rates will be 0, pitch and roll will be 0 if Tasks 5 and 6 were successful; otherwise, the angles are not predetermined.

Sequence 2, starting at DELELE (4A07) tests if the alpha error is within limits Y1 and Y2. The comments on pages 536 and 537 are misleading with regard to pass/fail conditions. Bit 6 of ANGR will be set if the test passes and delta alpha is greater than Y1. Bit 5 will be set by EL2FAL (4A49) if delta alpha is less than Y2. EL2FAL and EL2PAS are misnamed. This is not a program error, just a confusion factor. If delta alpha is within limits, measured roll angle rate output to the MDM will be set to all 1's; otherwise, it will be set to 0's. Pitch angle rate will remain 0. Range and range rate data-good flags will be the same as in sequence 1, angle and angle rate data-good flags will be as provided by the EA2, presumably 1 if the test is successful. Range, range rate, pitch and roll will be as in sequence 1. The self-test MUX is set to measure delta beta with POW5 set to 85 and OUTFLG and STWAIT are set. Sequence 3, the beta test, compares the measured delta beta with thresholds Y3 and Y4. Bit 4 of ANTR is set if the delta beta is greater than Y3 and bit 3 is set if less than Y4. Bit 2 is set if the angle track enable test is true. STPCNT is set to 8 for Task 9 and OUTFLG is set. Pitch rate will be set to all 1's if the test succeeds or all 0's if it fails. All other outputs will remain the same as in sequence 2.

4.9 Task 9

Task 9 (5409) is the receiver sensitivity test. This uses the test dipole at less than full power. During the first pass, the system is configured to GPC acquisition mode by the macro

5409 TASK9 CONFIG :10,:00.

Estimated range is set to 10,250 ft, angle and angle rate data-good flags are forced to 1, SEQUEN is set to 1, and STWAIT is set for the 10-sec pause. After the pause, a sequence equal to 1 causes a branch to TRKFLG (545E). If TRACK is true, bit 8 of RSTY is set. SEQUEN is set to 2 and the test target is set by loading hex CD into POW5. This sends the radar signal strength to the comparator, sets ATTEN1 = 1 and ATTEN2 = 0 for a nominal output of -45 dBm to the dipole. All MDM outputs will be the same as they were during the previous pause, except the mode flag will be GPC ACQ and angle and angle rate data-good flags will be 1's. Sequence 2 goes to SIGSTV (548E), which compares the signal strength with thresholds R1 and R2. If signal strength is greater than R1, bit 7 of RSTY is set, and bit 6 is set if signal strength is less than R2. POW5 is set to D5 to measure delta alpha error voltage, SEQUEN is set to 3, OUTFLG is set, and STWAIT is set. MDM output will be identical to that of sequence 1. Sequence 3 goes to ALPHA (54F4) to compare the delta alpha error signal with thresholds ALP1 and ALP2. If the error signal is greater than ALP1, bit 5 of RSTY is set, and bit 4 is set if the error signal is less than ALP2. POW5 is set to 4, OUTFLG is set, and STWAIT is set. Again, MDM output is identical to Tasks 1 and 2. Sequence 4 goes to BETA (5555) to compare the delta beta error with thresholds BETA1 and BETA2. If delta beta is greater than BETA1, bit 3 is set and, if less than BETA2, bit 2 is set. Range and range rate limits are tested using routines RNGPA5 and RRATE, the same routines used by sequence 3 of Task 7. Bit 8 of RSTYRR is set if the range is 10,240 ft ± 250 ft, and bit 7 is set if the range rate is within ±3 ft/s. STPCNT is set to 9 for TASK10, OUTFLG is set and STWAIT is set. MDM outputs will be identical to those of sequence 3, with the exception of range and range rate which will reflect the values measured by the EA2 using the lower power test target.

4.10 Task 10

Task 10 tests the flags set in the prior tasks and sets E1TEST, E2TEST, DATEST and SYSTST, accordingly. SYSTST is set if all tests were successful and resets otherwise. These four flags are output to the MDM in status word 3. Range data to the MDM and D&C is set to 100,000 ft if

all tests pass, or 1,000 ft if any test fails. During the first pass through task 10, MDM output is initiated with OUTFLG set to 1. LAMP is set, and subsequent passes through task 10 will set SCANNG, TRACK and SCWARN flags which, in turn, will light the search, track and scanwarn lights in the D&C panel. Task 10 will continue to be executed until self-test conditions are reset.

Table 4.2. MDM Output at Conclusion of Power Form Test
Tasks 1-3, Sequence 1

MDM Output	Pass	Fail
Mode	Existing mode prior to self-test	Same
Range Data Good	0	Same
Range Rate Data Good	0	Same
Angle Data Good	0	Same
Angle Rate Data Good	0	Same
Range	All 1's (2,621,440 ft)	Same
Range Rate	All 1's (1638.35 fps)	Same
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	Indeterminate	Same
Pitch	Indeterminate	Same

Table 4.4a. MDM Output at Conclusion of Servo Test
Task 5, Sequence 1

MDM Output	Pass	Fail
Mode	GPC Designate	Same
Range Data Good	0	Same
Range Rate Data Good	0	Same
Angle Data Good	0	Same
Angle Rate Data Good	0	Same
Range	All 1's (2,621,440 ft)	Same
Range Rate	All 1's (1638.35 fps)	Same
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0°	0°
Pitch	30°	Indeterminate
Comment		Go to task 7

Table 4.4b. MDM Output at Conclusion of Servo Test
Task 5, Sequence 2

MDM Output	Pass	Fail
Mode	GPC Designate	Same
Range Data Good	0	Same
Range Rate Data Good	0	Same
Angle Data Good	0	Same
Angle Rate Data Good	0	Same
Range	All 1's (2,621,440 ft)	Same
Range Rate	All 1's (1638.35 fps)	Same
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	-30°	Indeterminate
Pitch	30°	30°
Comment		Go to task 7

Table 4.4c. MDM Output at Conclusion of Servo Test
Task 5, Sequence 3

MDM Output	Pass	Fail
Mode	GPC Designate	Same
Range Data Good	0	Same
Range Rate Data Good	0	Same
Angle Data Good	0	Same
Angle Rate Data Good	0	Same
Range	All 1's (2,621,440 ft)	Same
Range Rate	All 1's (1638.35 fps)	Same
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0°	Indeterminate
Pitch	0°	Indeterminate
Comment		Go to task 7

Table 4.5a. MDM Output at Conclusion of Power Test
Task 6, Sequence 1

MDM Output	Pass	Fail
Mode	GPC Designate	Same
Range Data Good	0	Same
Range Rate Data Good	0	Same
Angle Data Good	0	Same
Angle Rate Data Good	0	Same
Range	All 1's (2,621,440 ft)	Same
Range Rate	All 1's (1638.35 fps)	Same
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0°	Same
Pitch	0°	Same

Table 4.6a. MDM Output at Conclusion of $R,\dot{R}$ Test
Task 7, Sequence 1

MDM Output	Pass	Fail
Mode	Manual	Same
Range Data Good	1	0 or 1
Range Rate Data Good	1	0 or 1
Angle Data Good	0 or 1	Same
Angle Rate Data Good	0 or 1	Same
Range	5,320 ± 250 ft	Indeterminate if range check fails
Range Rate	0 ± 3 fps	Indeterminate if range rate check fails
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same

PRECEDING PAGE BLANK NOT FILMED

Table 4.6b. MDM Output at Conclusion of $R,\dot{R}$ Test
Task 7, Sequence 2

MDM Output	Pass	Fail
Mode	Manual	Same
Range Rate Good	0	0 or 1
Range Rate Data Good	0	0 or 1
Angle Data Good	0	0 or 1
Angle Rate Data Good	0	0 or 1
Range	All 0's	Indeterminate
Range Rate	All 0's	Indeterminate
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Comment	EA2 to EA1 serial data forced to 0 unless tracking	

Table 4.6c. MDM Output at Conclusion of $R,\dot{R}$ Test
Task 7, Sequence 3

MDM Output	Pass	Fail
Mode	Manual	Same
Range Data Good	1	0 or 1
Range Rate Data Good	1	0 or 1
Angle Data Good	0 or 1	0 or 1
Angle Rate Data Good	0 or 1	0 or 1
Range	10,240 ± 250 ft	Indeterminate if range check fails
Range Rate	0 ± 3 fps	Indeterminate if range rate check fails
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same

Table 4.7a. MDM Output at Conclusion of Servo Angle Track Test
Task 8, Sequence 1

MDM Output	Pass	Fail
Mode	Autotrack	Same
Range Data Good	1	0 or 1
Range Rate Data Good	1	0 or 1
Angle Data Good	1	Same
Angle Rate Data Good	1	Same
Range	Should be close to 10,240 ft	Same
Range Rate	Should be close to 0 fps	Same
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Comment	Range and range rate as provided by test target	

Table 4.7b. MDM Output at Conclusion of Servo Angle Track Test
Task 8, Sequence 2

MDM Output	Pass	Fail
Mode	Autotrack	Same
Range Data Good	0 or 1 (same as sequence 1)	Same
Range Rate Data Good	0 or 1 (same as sequence 1)	Same
Angle Data Good	1	Same
Angle Rate Data Good	1	Same
Range	Should be close to 10,240 ft	Same
Range Rate	Should be close to 0 fps	Same
Roll Angle Rate	All 1's (-16.38°/s)	All 0's
Pitch Angle Rate	All 0's	All 0's
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Comment	Range and range rate as provided by test target	Same

Table 4.7c. MDM Output at Conclusion of Servo Angle Track Test
Task 8, Sequence 3

MDM Output	Pass	Fail
Mode	Autotrack	Same
Range Data Good	0 or 1 (same as sequence 1)	Same
Range Rate Data Good	0 or 1 (same as sequence 1)	Same
Angle Data Good	1	Same
Angle Rate Data Good	1	Same
Range	Should be close to 10,240 ft	Same
Range Rate	Should be close to 0 fps	Same
Roll Angle Rate	All 0's or all 1's depending on outcome of sequence 2	Same
Pitch Angle Rate	All 1's (-16.38°/s)	All 0's
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same

Table 4.8a. MDM Output at Conclusion of Receiver Sensitivity Test
Task 9, Sequence 1

MDM Output	Pass	Fail
Mode	GPC Acquisition	Same
Range Data Good	0 or 1 (same as task 8, sequence 1)	Same
Range Rate Data Good	0 or 1 (same as task 8, sequence 1)	Same
Angle Data Good	1	Same
Angle Rate Data Good	1	Same
Range	Should be close to 10,240 ft (same as task 8, sequence 1)	Same
Range Rate	Should be close to 0 fps (same as task 8, sequence 1)	Same
Roll Angle Rate	All 0's or all 1's depending on outcome of task 8, sequence 2	Same
Pitch Angle Rate	All 0's or all 1's depending on outcome of task 8, sequence 3	Same
Roll	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same

Table 4.8b. MDM Output at Conclusion of Receiver Sensitivity Test
Task 9, Sequence 2-4

MDM Output	Pass	Fail
Mode	GPC Acquisition	Same
Range Data Good	0 or 1	Same
Range Rate Data Good	0 or 1	Same
Angle Data Good	1	Same
Angle Rate Data Good	1	Same
Range	10,240 ± 250 ft	Indeterminate
Range Rate	0 ± 3 fps	Indeterminate
Roll Angle Rate	All 0's or all 1's, depending on outcome of task 8, sequence 2	Same
Pitch Angle Rate	All 0's or all 1's, depending on outcome of task 8, sequence 3	Same
Roll	0° of tasks 4 and 5 pass Indeterminate otherwise	Same
Pitch	0° if tasks 4 and 5 pass Indeterminate otherwise	Same
Comment	Range, range rate and range data good flags may differ from seq. 1 due to lower power	Same

Table 4.9. MDM Output at Conclusion of Flag Compilation
Task 10

MDM Output	Pass	Fail
Mode	Manual	Same
Range Data Good	0 or 1	Same
Range Rate Data Good	0 or 1	Same
Angle Data Good	1	Same
Angle Rate Data Good	0	Same
Range	100,000 ft	1,000 ft
Range Rate	Should be close to 0 fps (same as task 9, sequence 4)	Indeterminate
Roll Angle Rate	All 0's	Same
Pitch Angle Rate	All 0's	Same
Roll	0°	0° if tasks 4 and 5 pass Indeterminate otherwise
Pitch	0°	0° if tasks 4 and 5 pass Indeterminate otherwise
DA TEST	1	0 or 1
E1 TEST	1	0 or 1
E2 TEST	1	0 or 1
SYS TST	1	0

5.0 MEMORY USAGE OPTIMIZATION

5.1 Introduction

In the present hardware configuration, the RCA processor is not limited by available memory space. If feed-forward for the antenna servo is implemented, however, time considerations may play an important role in the processor's ability to perform the required functions. Initial estimates by HAC indicate a timing margin of 3-6% if feed-forward is implemented.

One way to improve the efficiency of the present software from the standpoints of both execution time and memory requirements, would be to redefine the multitude of flags in terms of one flag/word rather than eight/word. This has the effect of requiring more memory to store flags but effects a net increase in memory due to the simplified software required to test or modify flags. An additional benefit is that computational speed is increased.

5.2 Suggested Modifications

With the current implementation, one flag word may hold as many as eight flag bits. In order to access an arbitrary flag bit, assembler macro instructions are used which generate inline code as the program is assembled. These macro instructions are used to GET, SET, RESET or TOGGLE flag bits. For example, the instruction

```
RESETS  REGISTER,FLAG
```

is expanded to the following inline code:

```
LDI  (low-order address of flag word)
PLO  REGISTER
LDN  REGISTER
ORI  (flag bit to be reset)
XRI  (flag bit to be reset)
STR  REGISTER.
```

This macro uses nine memory words and 12 machine cycles, with the high-order byte of REGISTER preset to the high-order address of the flag word. By contrast, if one word contained one flag, the flag would be reset merely by clearing the word. This could be done with four instructions, versus six, using six memory words and eight machine cycles. Similarly, the SETS macro, which takes seven memory locations, five instructions and 10 cycles, could be implemented with five memory locations, four instructions and eight cycles.

The use of one-bit flag words would require about 120 additional memory words for flags; however, these would be gained back by the savings of memory in one routine, CONFIGURE, which makes extensive use of flags. Additional space would be required in input/output routines to format the bit flags; however, this can be done with relative efficiency since the flag access is not random as it is with the GET, SET and RESET macros. Parallel output word 5 (POW5) could remain parallel since all bits of POW5 are typically loaded at the same time.

While there is no point in modifying the existing software if it is adequate to do the job, in the event that feed-forward requires too much time, this technique could be tried. The effect of the suggested changes can easily be tested merely by redefining the macros, allocating storage for the flags and redefining the input/output bit mapping.

6.0 CONCLUSIONS AND RECOMMENDATIONS

The EA1 software is well documented and annotated, with the exception of the omission of a cross-reference listing in the documentation. In this report, Axiomatix has provided a cross-reference listing of status flags to partially fill this gap. A minor bug in the procedure termination routine has been found and the fix is discussed in section 1.1. Executive reconfiguration and self-test are examined and appear to perform as expected. Discussion of the MDM output during self-test is very brief in the HAC documentation; a more complete discussion of the MDM outputs is provided in this report. A change in the method of flag storage is recommended if processing time becomes a critical factor in future versions of the EA1 software.

REFERENCES

1. HAC Ku-Band EA1 Flight Software, Release IV (Listing), 17 August 1979.

2. HAC EA-1 Software Description Document, Releave IV, 17 August 1979.

3. HAC EA1 Preliminary Design Review, Document #HS237-1531, Vol. II, Part 1, March 20, 1978.

4. HAC EA-2 Preliminary Design Review, Vol. III, Document #HS237-1531-3, March 14-24, 1978.